关于嵌入式软件重构

一、嵌入式软件重构目的

由于嵌入式软件的通讯协议及识别逻辑方式还存在不足，譬如通讯协议中有些数据在传输中有些数据冗余、部分数据传输方式还有待改进（如某些特征值数据需要先写然后才能再读）。这些问题导致在通讯过程很消耗硬件以及软件的资源，并且会造成较高的功耗。

1. 改动方向

1、硬件的唯一标识方式由BLE的Service UUID改为由SSID作为唯一标识进行识别。

2、更改通讯协议模式，由少特征值多功能复用更改为多个特征值，一个特征值对应一个功能，减少数据长度及冗余数据量。

3、更改代码执行方式，优化代码，提高执行效率。

1. 协议具体改动初步设想

1、数据格式表格

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 特征值UUID | 功能 | 长度  （字节） | 性质 | 格式 | 读/写时间间隔(ms) |
| 信息读取相关指令 | FFF0 | 硬件信息 | 4 | 读/写 | 桌腿类型 硬件类型 版本号\_H 版本号\_L | - |
| FFF1 | 存入/读取预设高度（4个） | 8 | 读/写 | 预设高度0\_H 预设高度0\_L 预设高度1\_H 预设高度1\_L 预设高度2\_H 预设高度2\_L 预设高度3\_H 预设高度3\_L | - |
| FFF2 | 站坐姿态判断阈值 | 2 | 读/写 | 判断阈值\_H 判断阈值\_L | - |
| FFF3 | 桌子动作上下限 | 4 | 读/写 | 动作阈值上限\_H 动作阈值上限\_L 动作阈值下限\_H 动作阈值下限\_L | - |
| 控制相关 | FFF4 | 控制指令 | 1 | 写 | 指令 | 500 |
| FFF5 | 高度及状态 | 4 | 读/通知 | 桌子状态 人体姿态 信息/高度\_H 信息/高度\_L | 500 |
| 锁定状态相关 | FFF6 | 锁定状态+延时锁时间 | 5 | 读/写 | 锁定状态 时间\_03 时间\_02 时间\_01 时间\_00 | 1000(读) |
| 健康数据相关 | FFF7 | 开机时间戳 | 4 | 读 | 时间\_03 时间\_02 时间\_01 时间\_00 | 1000(读) |
| FFF8 | 健康数据(201组) | 16 | 通知 | 包计数 (姿态 时间\_03 时间\_02 时间\_01 时间\_00) (数据包2) (数据包3) | 30 |

关于各个特征值数据的解释：

FFF0：硬件信息（存入snv）

手机端通过读取特征值FFF0可以得到当前设备的硬件信息，这些硬件信息全部都写在非易失性存储器中。断电后，硬件信息依然保存在设备中。由于现有版本的设备无法识别桌腿的类型，所以加入写的属性可以通过APP来设定该设备的桌腿类型。

硬件信息包括：桌腿类型（包括控制盒类型）、设备类型（Cube\Bar）、具体版本信息。

FFF1：高度及状态

手机端可以直接通过读取该特征值的数据来得到当前桌子的信息，包括：高度值、桌子信息、人体姿态信息、以及桌子错误信息。而高度值 = 高度高位x256 + 高度地位（十进制）。具体的错误信息以及人体姿态信息见下表。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能名称 | Byte0 | Byte1 | Byte2 | Byte3 | 备注 |
| 桌子正常下的位置信息以及人体姿态 | 0x01 | 0x00 | Height\_H | Height\_L | 桌子正常，人体离开 |
| 0x01 | 桌子正常，人体姿态坐 |
| 0x02 | 桌子正常，人体姿态站 |
| 异常状态信息 | 0x02 | - | 0x01 | 0x00 | M1过流E01 |
| 0x02 | - | 0x02 | 0x00 | M2过流E02 |
| 0x02 | - | 0x04 | 0x00 | M3过流E03 |
| 0x02 | - | 0x40 | 0x00 | M1霍尔信号断E07 |
| 0x02 | - | 0x80 | 0x00 | M2霍尔信号断E08 |
| 0x02 | - | 0x00 | 0x01 | M3霍尔信号断E09 |
| 0x02 | - | 0x00 | 0x10 | 工作制保护H01 |
| 0x04 | - | 0x01 | 0xaa | 复位RST |
| 手控器存储信息 | 0x06 | - | 0x00 | 0x00 | 未存储状态S-，持续8秒 |
| 0x06 | - | 0x01 | 0x00 | 存储S-1，持续3秒 |
| 0x06 | - | 0x02 | 0x00 | 存储S-2，持续3秒 |
| 0x06 | - | 0x04 | 0x00 | 存储S-3，持续3秒 |
| 0x06 | - | 0x08 | 0x00 | 存储S-4，持续3秒 |

FFF2：站坐姿态判断阈值（存入snv中）

该特征值用于更改设备的站坐判断阈值，应用于某些特殊的场合，比如身高比较特殊的用户以及不同单位制的控制盒。

注：该值也可以从设备读到APP中。

FFF3：桌子动作上下限（存入snv中）

该特征值可以设置/读取桌子动作的上限/下限。格式为：动作阈值上限\_H 动作阈值上限\_L 动作阈值下限\_H 动作阈值下限\_L。

其中 真实高度数据 = 数据\_H \* 256 + 数据\_L 。

在设置桌子动作上下限时，如果数据值为0，则忽略该数据的设置。

FFF4：控制指令

控制指令分为多种指令：

（1）基础指令：上升、下降、停止

（2）到达预设高度指令：预设0、预设1、预设2、预设3

（3）进阶指令：获取健康数据、等等……

指令表格如下：

|  |  |  |
| --- | --- | --- |
| 分类 | 指令 | 功能 |
| 基础指令 | 0x00 | 停止 |
| 0x01 | 上升 |
| 0x02 | 下降 |
| 预设高度指令 | 0x10 | 动作到预设高度0 |
| 0x11 | 动作到预设高度1 |
| 0x12 | 动作到预设高度2 |
| 0x13 | 动作到预设高度3 |
| 进阶指令 | 0x20 | 获取健康数据 |

FFF5：存入/读取预设高度（4个 也存入snv中）

预设高度一共4个，通过该特征值可以修改预设高度，也可以读取预设高度，在写入预设高度时，如果设置的高度值为0，则忽略该预设高度的设置。

格式为：预设高度0\_H 预设高度0\_L 预设高度1\_H 预设高度1\_L 预设高度2\_H 预设高度2\_L 预设高度3\_H 预设高度3\_L

其中：预设高度值 = 数据\_H \* 256 + 数据\_L

FFF6：锁定状态+延时锁时间（锁定状态存入snv中而延时锁时间不存）

该特征值用于读取剩余非解锁状态时间及状态/设置当前锁状态。设置的时间为从当前时间算起，剩余的非解锁时间。

格式为：锁定状态 时间\_0 时间\_1 时间\_2 时间\_3

如果设置的锁定状态为0x01的条件下将时间戳设置为0x00 0x00 0x00 0x00的话，那么桌子就会立即进入锁定状态，在锁定状态为0x01的条件下，如果设置时间戳为非零的话，那么桌子进入延时锁定状态，剩余解锁时间为设置的时间戳的值。

如果设置的锁定状态为0x00的话，那么无论后面的时间戳设置时什么，桌子都会进入到永久解锁状态。

FFF7：获得开机时间戳

该特征值用于得到硬件自上电到当前时间的时间戳。内部数据每一秒更新一次，APP只需要读取该特征值内的数据即可得知硬件当前的上电时间戳。

FFF8：健康数据（201组）

该特征值用于发送健康数据，健康数据最多可以在底层存储201个包，每个包有一个包计数数据和三组数据，每组数据5个字节（姿态 时间 时间 时间 时间）。现阶段它存储在已逝性存储器中，断电后便会消失。未来或许可以存储在非易失性存储器中。

在控制指令中写入0x20时，FFF8开始发送健康数据，每30ms发送一个包，一共201个包。